Articulating the speed(s) of the Internet: the case of open source/free software

Holtgrewe, Ursula

Postprint / Postprint
Zeitschriftenartikel / journal article

Ursula Holtgrewe

Articulating the Speed(s) of the Internet: The Case of Open Source/Free Software

Paper contributed to the KWI conference "Speeding up Cultural Change" June 26 – 29, 2002 Mülheim

Abstract:

The Internet is widely considered as a key factor of speeding up social and cultural change. It represents the merging of information and communication technologies and enables flows of information and capital, and communication and co-operation regardless of space and, possibly, time. The paper explores the example of *Open Source/Free Software* development, i. e. software development in self-organised projects based on a considerable share of voluntary work. Here, we find complex articulations of speeding up and slowing down technological development. *Open Source/Free Software* projects complement the logic of speeding up technological progress and of obsolescence with a reflexive logic of optionality, variety and sustainability which addresses the accessibility of technology and knowledge as a precondition for future creativity beyond markets and organisations.

Keywords:

Software development – information technology – creative action – sustainability – flexibility

Time and the Internet: Media and Tools

While there is wide agreement in the sociology of technology that technology is genuinely social and embedded with social action in both enabling and restricting ways, theories on social time and the Internet still can be grouped according to their respective emphasis on the technology's empowering or corrosive effects. In his influential "Network Society", Manuel Castells (1996) argues that information and communication technology especially has fundamentally changed social structures and processes, times and spaces. This is not meant in a technologically determinist sense. Castells is very attentive to the social and institutional prerequisites of these technological changes, but they appear to have taken off in an historically unprecedented way. Information and communication technologies address the generation and application of knowledge directly and (together with genetic engineering) they lead to a new technological paradigm: these technologies are recursively self-enhancing. Their new quality is

> "the capacity of these technologies to self-expand their processing power because of the feedback on technological development of the knowledge generated on the basis of the

technology" (Castells, 2001).

For this reason they come to play a key role in a fundamental re-organisation of social relations in and through networks. This has far-reaching effects on social times and temporalities. In general, the circulation of capital and knowledge compresses time and space, which leads companies and investors to *manage* time in new, relational ways:

> "Time is managed as a resource not under the linear, chronological manner of mass production, but as a differential factor, in reference to the temporality of other firms, networks, processes or products" (1996: 439).

The changing patterns of the economy and of work then lead to a fluid dynamic of projects and networks. Previous temporalities erode, and this relates to both the linear and sequential time of modernity and the rhythmical times of the body and of social life. Somewhat surprisingly, this leads Castells to an argument of social de-differentiation: The Internet constitutes a "timeless time" which de-differentiates social and cultural temporalities, immediacy and eternity (464ff.).

The philosophers Hubert Dreyfus and Fernando Flores (1997) take the argument of de-differentiation a critical step further. They draw on the work of Danish philosopher Soren Kierkegaard to point out the corrosive effects all of this may have on human agency. Interestingly, Kierkegaard argued around 1850 that the emergence of the press and the early modern public sphere created a very similar immediacy and arbitrariness of information, eliminating criteria of relevance and significance in the process. It produced a particular configuration of subjectivity: an anonymous, detached spectator who is curious and interested in anything but not involved with anything. Meaningful distinctions demanding decisions, actions and taking responsibility disappear. Consequently, the groundings for action in the world, for taking risks and also for learning and ethical self-positioning erode.

With information arranged through hypertext links on the web, Kierkegaard's gloomy observations come to be fulfilled in Dreyfus's and Flores's view:

> "the user is no longer a subject who desires a more complete and reliable model of the world but a protean being ready to be opened up to ever new horizons" (Dreyfus/Flores 1997: 4).

If surfing the web becomes a way of life it

> "produces a self that has no defining content or continuity but is open to all possibilities and to constantly taking on new roles" (7).

This technologically enhanced openness of the self, however, is where other authors such as Sherry Turkle (1995) or Pekka Himanen (2001) observe increased possibilities for learning and experimentation. Virtual spaces may be used to construct, deconstruct and play with

identities. This “avant-gardist“ line of thought, which focuses on empowerment, learning and experimentation, has a tradition in the media theories of f. i. Brecht (1932) and Benjamin (1934) and in situationist anarchism (Debord, 1995). Brecht and Benjamin argued that then new technologies, i. e. radio and film, would enable the socialisation of expertise and unfold the creative potential of the masses. Recently, authors such as critical management theorist Paul Adler (2001) or radical sociologist Maurizio Lazzarato (1998; 2002) are reviving this line of argument: skill upgrading, knowledge work and information and communication technologies may unfold the forces of production in the hands of the new knowledge and creative workers.

It is well worth noting the cyclical return of both lines of argument in the face of new media. The press, film, radio, television and the Internet have been assumed to have similar, either corrosive or liberating effects on subjectivity and social development. Apart from Kierkegaard, authors in the line of critical theory such as Horkheimer and Adorno (1971) explored the corrosive effects of cultural change, new communication technologies and artistic forms on subjectivity and agency. Their conclusions are similar to the recent ones of f. i. Zygmunt Bauman (1997) or Richard Sennett (1998) who address changes in culture, consumption and work and their effects on subjectivity.

With a focus on the technology rather than users' subjectivity, there is some controversy over the Internet's status as a medium of collective memory as well. Systems theorist Elena Esposito argues that the Internet fundamentally changes the relationship of social memory and social forgetting:[i] While a library or an archive-cum-catalogue embodies a notion of bodies of knowledge being stored for a society or community, the Internet-cum-search engines is more performative and selective: it embodies the potentiality of producing new information by linking possible informations (Esposito, 2002). New information is generated from connections between selections, “memories which have never been thought before” (358, translation U. H.). This interactivity, however, is based on the medium's assumed superficiality: The machine renders itself invisible behind (user or programming) interfaces, so that the net as a medium of memory comes to resemble ancient divinatory memory systems rather than the mass media and archives of modernity.

On the other hand, modernist thinkers such as innovation theorist Ilkka Tuomi focus on the transparency and capacity of the Internet as a repository of context-specific tools to enhance collective memories: ”On the net we live in dog years, but our memory is that of an elephant” (2002). Using open source software development as a key example of collective and distributed innovation, he points out that “in open source, black boxes have transparent and

penetrable walls" (7). While complexity is of course reduced through all available technological means, we shall see that such black-boxing is deliberately kept reversible.

It thus appears that the established distinction of a medium or a machine/tool perspective on computing (Esposito, 1993) continues to organise the respective lines of argument. A "media" perspective tends to focus on the consumption and "surfing" side of Internet (or rather world wide web) use, and thus leans towards either postmodernism or critical theory. A modernist "tool" perspective foregrounds the recursively productive and innovative aspect of f. i. software development – and avant-gardist theorists fit into the tool perspective by seeing media as tools of expression and liberation.

Between Corrosion and Liberation: Agency and Practice

From either point of view, the technology opening up virtual spaces and vast realms of knowledge (to those who have access to it) is seen to fundamentally change human agency. Human agency is supposed to be either undermined or liberated, but either way it is no longer tied to socially embedded rhythms of living, to a biographically constituted ordering of experience and to a standpoint from which to develop plans and visions reaching into the future. However, if we understand technology as intertwined with human agency in ongoing processes of mutual shaping, actualisation and delegation (Rammert, 1997), the question is less one of technological effects but rather one of old and new articulations of agency, time and technology. Here, Castells points in a promising direction of theoretical and empirical enquiry as well:

> "the whole ordering of meaningful events loses its internal, chronological rhythm, and becomes arranged in time sequences *depending upon the social context of their utilisation*" (Castells 1996: 462, italics added),

and this precisely is the point. If given structures and rhythms erode, actors in these contexts need to reconstruct some kind of structure and rhythm. Just any kind of knowledge and information may be out there on the Net immediately and eternally, but it still needs to be found, understood and selected. More precisely, in order to become knowledge that matters, information needs to pass through human brains and it has to be made sense of both individually and collectively.

This is also the reason why recent discussions of knowledge work emphasise the relational and contextual character of knowledge. If knowledge is to make a difference in social action, it is embedded with social relations, and it is continuously de- and recontextualised. People need to navigate and to order information in order to make sense of it and be able to act upon it – and they all too frequently need to cut these processes short as well. It is no coincidence

then that in the network society, concepts such as situated learning, communities of practice (Lave/Wenger, 1991), *ba* (Nonaka et al, 2001) etc. emerge and gain influence. All these concepts emphasise the localised and situated character of knowledge and seek to explore and enhance its mobilisation and circulation by creating spaces and communities for such circulation. Complementary to the decontextualising powers of information and communication technology, the social, frequently face-to-face and craft aspects of knowledge creation and utilisation are being rediscovered by innovation researchers and also by management practitioners – and open source/free software is frequently cited as a prime example of such practices (Moon/Sproull, 2000; Edwards, 2001; Tuomi, 2002).

Sensemaking, however, is an inevitably sequential activity since human attention is limited (Weick, 1995).[ii] The very abundance of simultaneously available information requires its temporal ordering: establishing relevance, priority or urgency, keeping up to date or choosing which information to ignore, are activities requiring reflexive monitoring by individuals and groups. The simultaneity of networked information and activity then can be expected to be structured into sequences that make sense. Such context-specific structurations draw on situated action, but also on culturally and institutionally established patterns of sequences: projects, narratives or the notion of priority in science and technology are such patterns of temporal ordering, which can be found in the new contexts of Internet communities as well. The rules of 'netiquette' (i. e. the etiquette of electronic communication, f. i. http://www.dtcc.edu/cs/rfc1855.html) present norms of efficient and economical use of both technical bandwidth and human attention in online communication. And of course, information technology itself can be used and developed to support (and in turn, shape) the establishment of temporal orders and sequences.

Approaches addressing effects of "the" technology on societies as a whole thus presume too much uniformity of trends and linearity of causes and effects, if they emphasise networks, virtuality etc. under the perspective of de-differentiation. The notion of contextuality, however, fits in with Hanns-Georg Brose's (2002) concept of an increasing "simultaneity of the non-simultaneous" (cf. Bloch, 1962) which draws attention to a *differentiation* of time horizons and social temporalities. It addresses societies', collective and individual actors' "growing sensitivity for path-dependencies and irreversible, time-binding effects of decisions" (Brose 2002: 1). This perspective makes it possible to address questions of agency and social time in terms of optionality and reflexivity. It is open to the complex and heterogeneous articulations of knowledge, technology, actors and institutions in and over time.

A useful theoretical framework is offered by neo-pragmatist social theory. Emirbayer and Mische (1998) f. i. conceptualise agency and its structural contexts as both temporally embedded and constitutive of time. Actors then act in the present, define and accommodate problems and solutions, contingencies and strategies. While doing this, they continuously draw on past experiences and habits, and anticipate and design future outcomes and options. Action thus reflects back onto its temporal dimensions: Past history may be rewritten, alternative futures may move into the picture. Actors' accounts, their visions and their stories are changing through the course of action they embark upon. The relationship of agency and its dimensions thus needs to be seen as one of recursive and interrelated loops in which action itself constitutes different temporalities.

Through the example of free software/open source we are going to explore such articulations of past, present and future in the field of information technology which make it clear that temporally constituted agency continues to matter in this field: to initiate projects, reflect upon their preconditions, to co-ordinated distributed work, and to sustain creativity beyond the logic of the market.

Free Software/Open Source

Open source or free software[iii] is software developed in non-commercial, voluntary projects in which a number of developers (ranging from one person to hundreds or even thousands of people) create, test, improve, document and maintain computer programs and modules of programs. Computer programs generally are written in higher programming languages and then compiled, i. e. translated into machine language to run on a computer (compiled). With proprietary software, typically, the compiled version is sold, while open source programs come with the source code in order to be developed further. Open source projects describe themselves as decidedly open in a social sense as well: everybody who is able to deliver a qualified contribution is invited to do so. Discussion takes place over public mailing lists and newsgroups. In recent years, platforms such as Sourceforge or Berlios (www.sourceforge.org, www.berlios.de) have been established which provide forums and overviews over projects, offering services such as communication channels, noticeboards, version administration etc.

Open source/free software does not simply constitute a departure from the established worlds of commercial software development. It continues traditions of public goods and of the open academic circulation of knowledge which were present in the early development of the Internet before software even became a separate commodity (Holtgrewe/Werle, 2001). The self-referential character of the Internet, with the technology providing the medium for its further innovation has been based on a multitude of innovation traditions, and open source

projects were closely tied to its development. Here, heterogeneous actors and logics of innovation came together: the academic tradition of sharing, collaboration and knowledge generation as a public good, a military logic of decentralising infrastructure in order to decrease vulnerability, business interests of the developing computer industry and hackers' counter-cultural values of openness and creative self-expression (cf. Castells, 2001b).

Getting started: Plausible Promises, Charisma and Nostalgia

According to the open source theorist and spokesman Eric Raymond, projects typically start "by scratching a developer's personal itch" (Raymond, 1998), by someone starting to solve an actual problem s/he comes across, publishing their projects (i. e. descriptions and the source code), and inviting contributions and feedback. In order to start both on collective and individual projects, obviously available resources, skill and especially time are required. Whoever is able to design and initiate a project, articulates competencies, infrastructures, code and social capital. In order to attract contributions initiators need to give a "plausible promise" of usable and/or interesting results or new and interesting problems (Raymond 1998). This also means that a project should not be too "finished" already and still offer challenging problems. To get started then, both an itching problem, and frequently a certain megalomania and underrating of difficulties ("how hard can it be?") help (Moody, 2002). Then, in a way comparable to Weberian charismatic leadership (Turner, 1995; Blutner et al., 1999), spaces of possibility are opened up, people surprise themselves and abandon themselves to the process.

It is interesting to take a look at Linus Torvalds's famous 1991 invitation to collaborate on the operating system kernel which was to become Linux. Torvalds then was a 21 year old student of computer science at Helsinki University and looking for a unix-compatible operating system for his 386 personal computer. He posted the following to a newsgroup dedicated to Minix, another small operating system used in educating students:

> `Do you pine for the nice days of Minix-1.1, when men were men and wrote their own device drivers? Are you without a nice project and just dying to cut your teeth on an OS [operating system] you can try to modify for your needs? Are you finding it frustrating when everything works on Minix? No more all-nighters to get a nifty program working? Then this post might be just for you." (posting on Usenet-Newsgroup comp.os.minix, quoted in Moody 2002, 45)

Self-ironically Torvalds draws on the hacker image – a culture of intrinsic fun-oriented programming taking place outside and beyond everyday life. Immature programs, problems, bugs and lacks of device drivers are presented as masculine challenges which are first nostalgically conjured up (Strangleman, 2002) and then recreated.

Temporalities outside the everyday come into play on several levels. Nostalgia is conjured up, but in the fast-paced world of information technology it frequently reaches back only a couple of years. Besides, computer programming apparently is one of those creative activities which require intense concentration and may generate experiences of absorption and „flow“ in the process (Krafft/Ortmann, 1988; Csikszentmihalyi, 1991). The classical hacker image then conjures up a neglect of normal life, sleep, meals etc. in favour of obsessive programming, although empirically nowadays hackers and programmers seem to lead a more normal life of busy professionals trying to find the time to concentrate (Strübing, 1993). Yet, in a recent survey of FS/S developers (Lakhani et al., 2002) 72,6% of 684 respondents said that they frequently or always „lose track of time when programming“. Moody (2002: 47) reports anecdotal evidence that frequently initiators of FS/OS projects (such as Linus Torvalds himself) took important steps through intense bouts of programming over Christmas – young men escaping the boredom of family holidays by doing something absorbing outside the everyday. Self-determined programming then offers interest, challenges and chances for recognition outside the world of face-to-face connectedness.

This evidence suggests that only part of the attraction of FS/OS projects can be attributed to charisma in the sense of a charismatic relationship of leaders and followers. A considerable part of the charismatic motivation to try new things and get involved comes from the projects, the practice of programming, the code and the technology. It has a foundation in the subject matter. And at some time, decisions are taken, ideas evaluated, contributions come in and projects take shape.

FS/OS Projects: Simultaneity, Modularity, Loose Coupling

If the published project catches on and attracts users and contributors, it will be tested, ported to new technology, and features will be added. Founder-projects initiated by one person's problem-solving are, however, only part of the field of FS/OS. There are also projects initiated by small groups, and there are entire programs released as open source by companies in order to draw on external developing resources or to attract customers to their specification and support services. With larger projects, usually informal divisions of labour form. The founder keeps a central position and core teams and responsible maintainers of parts of the project emerge who select contributions and decide on new versions. The testing and further development of an innovation thus is informalised. It does not happen within an organisation or network with circumscribed boundaries but involves or tries to involve a large *potential* community out of which contributors self-select. Indeed, Benkler (2002) suggests that such “commons-based peer production” can be more economically efficient than either hierarchies

or markets – for the reason that individuals who volunteer for particular tasks have better information on their capabilities and fitness for a task than either firms or market contracts are able to specify.

Then, the wide range of user/developers means that testing and bug-fixing, a lengthy and frequently torturous process in commercial software development, can happen faster, in a wider range of utilisation contexts. Improvements are made continuously. The early and frequent releases of new versions are expected to make maximum use of a wide and distributed capacity of user/developers. Parallel and competing work is possible, but parallel effort will be limited by the frequency of new releases in which solutions and improvements are taken up as quickly as possible. The distributed and loosely co-ordinated efforts and contributions are made possible by the modular structure of software in the Unix world.[iv] It enables the simultaneous but independent development of small modules, but requires clear definitions of interfaces and programming standards.

However, there is evidence that contributions to open source projects are typically rather more unequally distributed than the normative idea of a "bazaar" (Raymond 1998) suggests. Ghosh, Robles and Glott (2002), analysing roughly five billion bytes of FS/OS code authored by 31999 developers, found that the most active 10% of developers had contributed 74% of the code. Projects (n= 16905) involved an average of 5,1 authors with the median at 2 authors, so that the majority of projects is indeed small. Projects thus are rather more modular and loosely co-ordinated than the possibilities of world-wide virtual co-operation seem to suggest (cf. Edwards 2001). The bazaar of potential exchanges materialises in a loosely-coupled and individualistic way.

Yet open source projects are not indifferent to waste of time and effort. They keep inventing technical and social mechanisms to manage the process: There are explicit standards and guidelines for coding, elaborate routines for submitting bug reports, and all of these of course are automated where possible. Version control systems such as CVS (http://ccvs.cvshome.org) allow programmers to work simultaneously on the same body of code, guaranteeing that they download the respective latest version to work upon. In this way, they also allow for temporary and flexible involvement. When the work is complete and has been tested, it is checked back into the main repository. Previous versions are kept so that changes are reversible and mistakes can be traced to their original appearance and eliminated (Iannacci, 2003). Version control systems thus represent specific organisations of a collective memory, which ties into the social world through relationships of collaboration and/or through technologically embedded hierarchies of rights to collect and commit changes to the

main repository. While they make sure work is kept current and redundancy is avoided, past versions are kept and may be gone back to just in case.

However, with all the modularity and openness, time lags in projects and work overload of maintainers can still be a problem which has frequently lead to crises. World-wide virtual co-operation still faces the problems of people's limited time and attention and, concurrently, of relevance. The openness and interchangeability of inputs does not lead to projects being indifferent to the uncertainties of "living labour", even though it permits a considerable amount of flexibility.

Flexible Involvement and Simultaneity

This view is supported by the results of a couple of recent developer surveys. They shed some empirical light on developers' demographics, involvement and motivation (Robles et al., 2001; Ghosh et al., 2002; Lakhani et al., 2002).[v] Results are fairly consistent: developers are youngish with an average age below 30 years. They are almost exclusively male (98 – 99%). 60 – 70% are university or college graduates, 20 – 30% are students. Around 80% are IT professionals, which leaves roughly a fifth of amateurs in the sense that they have nothing to do with the IT industry, which still is a considerable share (Ghosh et al., 2002). Between half and 80% of FS/OS developers are volunteers. For the majority, involvement is limited to the extent of a more or less time-consuming hobby. Roughly two thirds of developers spend less than 10 hours per week on FS/OS development.

However, there are indications of some interesting grey zones in between work and non-work involvement. Apparently, people develop FS/OS at work even though they are not directly paid for it (12,8% in the sample of Ghosh et al., 2002). 17% of Lakhani et al.'s (2002) sample develop FS/OS at work, though their supervisors do not know about it. For another 9% who do, it is not their core job. This suggests that FS/OS projects benefit from the discretion highly skilled computing experts have over their activities at work: an informal subsidy of FS/OS development by companies which is reminiscent of de Certeau's (De Certeau, 1988) concept of *la perruque* – workers using companies' resources for their own purposes during working time (cf. Tuomi, 2002: 28). However, this does not have to be illegitimate. A lot of open source programs are tools for software development and the running of computer systems so that work on these tools actually may make people's core jobs easier as well. Simultaneity comes in here in a different way: A local bug-fix or problem solution at someone's workplace can simultaneously become an FS/OS contribution without taking anything away from the company. In return, the FS/OS communities (and other relevant newsgroups or real-life communities of practice) become free sources of help and problem solving for workers and

their employers. This is made possible by the characteristics of knowledge goods: They can only be appropriated to a limited extent, and they can be simultaneously used and given away. A concrete solution at work does not lose its local value if is contributed to an OS/FS project – if and where companies do not have an interest in the commodification of just any solution or idea coming up in the process of work.

The evidence on time and involvement then suggests that the ideal-typical hacker lifestyle of excessive programming and absorption at the expense of everyday life is not really typical. The computer professionals and knowledge workers involved in FS/OS give the permeable boundaries of work and life a different accent. While commercial software development and similar occupations are characterised by flexible and extensive working times and project work (Perlow, 1998; Kalkowski/Mickler, 2002), open source developers find opportunities to gain time and space for non-profit work or a more or less subversive conversion of work into public goods and informalised co-operation.

Additionally, and probably to a larger extent, developers draw on a gendered use of leisure time, which may help to explain why the scene is even more predominantly male than computing in general. Pursuing time-intensive and absorbing hobbies is much of a masculine thing, and the sociology of gender and technology (Hacker, 1989) suggests that the specific passion for abstract technical excellence and bonding around it is developed in masculine rather than feminine socialisation. Social exchanges and groupings around technology then are likely to be male-dominated and if they are, they can easily establish a path-dependence of not attracting women.[vi]

All in all, developers' attitudes, practices and motivations look rather less heroic and charismatic than journalistic accounts (f. i. Moody, 2002) and also programmatic statements suggest. Yet FS/OS development has both a normative and an intrinsic side which contrast with the image of knowledge workers as driven by purely entrepreneurial or technical-instrumental motives. A focus on learning, the sharing and exercising of skill, a commitment to variety and freedom can be considered as use-value oriented and quasi-professional values with elements of fun and masculine hobbyism. They are reflexive if and when they address the conditions of creative production, and may even be socially transformative in the sense that they address and seek to create its social and institutional prerequisites as well.

Prospectivity and the Distribution of Options: Licences

With its character of voluntary, non-profit projects in the field of technology FS/OS negotiates the boundaries of markets, commodities and public goods respectively, i. e. the institutional conditions of technological innovations. Licences and intellectual property matter

in ways different from other innovation models, since in order to attract voluntary contributions, the possibilities of private appropriation must be limited. In the creation of open source licences, the focus is on social transformation in the direction of freedom, learning, the use-value and intelligent and co-operative use of products. The social innovation which focuses this transformative aspect is the so called copyleft (cf. www.gnu.org). This is the legal construction of a licence, the General Public License (= GPL) under which much open source software, also Linux, is distributed. It does not just rule that products are free but allows free distribution under the condition that further developments and applications are put under the same licence. The point of the GPL thus is its "infective" character. It is a tricky construction which uses the legal instruments of copyright to subvert it.

Although this very hacker-like paradoxical construction is amazing legally and socially, it has not become the standard for open source communities. The much-debated and very specialist issues of licencing (Working Group 2000) may be interpreted as negotiations over the temporal distribution of options. Copyleft ("All rights reversed") excludes private appropriation of intellectual products and ties a creative posterity to the conditions of open use – an attempt to tie oneself and others to freedom and variety. Other licences give copyrightholders and future developers the choice over licences, and even commercialisation – and thus restrict openness to the present.

The GPL however, does not appear to be just another utopian idea which has failed in the face of reality. The social innovation of the GPL together with the success of open source as a development model for high quality software have set a normative maximal standard and challenged standards and institutions of intellectual property. It has also, quite practically, limited the risks of defection and private appropriation of collaborative efforts. It mobilises social imagination and connects it with technological possibilities. Not least, it addresses the values a knowledge society places on creativity and access beyond the market.

Discussion: Success and Risk

Like any other innovation, software development requires a range of resources: knowledge and expertise, time, access to the technology. Software development is distinct from other fields of research and development in several ways. The traditional view (and self-image) of software development as a craft or even an art (Knuth, 1972) has been corrected by evidence of its stratification and rationalisation in commercial industries (Beirne et al., 1998). Yet, it still counts as a prime example of knowledge work, "a highly interpretive process" (Beirne et al., 1998: 152) of creative problem-solving by applying, de-and recontextualising knowledge. The result is "running code," i.e. functioning software. Since programming is a highly

complex activity in itself and through its interpretive and communicative side, programmers' productivity is notoriously hard to measure and varies widely. Time and effort can only roughly be calculated in relation to results. However, commercial projects need to do precisely that, and consequently are driven by deadlines, unexpected difficulties and unclear specifications (Kalkowski/Mickler 2002). Planning then works or doesn't in spite of itself, with deadlines both adding pressure and giving orientation, and also relieving uncertainty over quality.

By means of being outside the market, FS/OS projects modify the risks and uncertainties of software development in specific ways: since work is unpaid and products are not sold, productivity does not matter in an economic sense. Risk consists in wasted time and effort if contributions are not taken up or developments not continued. Yet such risk is limited if programming is valued intrinsically. Then the time has been spent satisfyingly anyway. In this way, open source development, like other hobbyist fields of activity, draws on a non-economy of flow and absorption in the process in which time is not really “spent“ at all.

Since the source code is available, and has hopefully been written in a transparent way, work can be divided and passed on in a flexible way, even though bottlenecks occur. And if a project is abandoned it is still available to be taken up again. This leads f. i. Martin Schulze of the debian/GNU project (a non-commercial Linux distribution) to claim that open source projects cannot really fail at all (http://www.oekonux.de).

FS/OS development projects then are successfully negotiating and limiting the risk of innovation by operating at a distance from market pressures and constructing a “safe” or “failure-friendly” environment for technological innovation. In this context, flexible time use and limited involvement, simultaneous and loosely coupled activities are both socially and technologically possible.

However, it would be overoptimistic to claim that FS/OS has *solved* the problems of highly skilled virtual co-operation. They are based on specific prerequisites: the modular structure limiting the needs for day-to-day co-ordination, the subject matter allowing a considerable base of consensus what “running code” is, the available talent meaning people being sufficiently interested in programming and finding the time, space and equipment to do so. While the communities pride themselves on their ethnic, cultural and professional diversity, the predominance of educated younger men suggests that such resources are not equally distributed through societies.

Conclusion: Sustaining Creativity

Not surprisingly, FS/OS development neatly fits into the "tool-oriented" empowerment perspective on the Internet. Here, the Internet does not present an undifferentiated mass of simultaneous and arbitrary information in "timeless time" (Castells 1996), nor is Esposito's vast potentiality of hypertext information only accessible through quasi-divinatory techniques. Even if such all-encompassing simultaneity erodes human capacity for action, and in other contexts and practices of Internet use such processes may well be found, in our specific context creative action has come out ahead: Meaningful action and normative commitments are not drowned in the endless possibilities of unlimited information and communication. These are balanced by the projects' focus on technical functionality and by the social norms and technological tools of virtual co-operation. The eternity and immediacy of Internet-based information and culture provides FS/OS hackers with a safe environment for code and ideas and with the means of co-operation which is selective and modular. The tools of f. i. mailing lists or version administration support non-simultaneous and flexible involvement as well as bouts of intensive hacking, and the specificities of knowledge production allow for simultaneous uses of time for work and the production of public goods – a certain, and possibly surprising time-sovereignty of OS/FS developers across the boundaries of working and non-working time. Both social and technologically supported selectivity and modularity thus work to reduce the overcomplexity of world-wide simultaneous co-operation and render it manageable in a sequential but loosely coupled way. This potential simultaneity is thus broken down into non-simultaneous modules and projects, but these, through an ensemble of tools, practices and decisions are articulated to result in coherently functioning code.

However, the case points beyond the focus on technological tools and solutions, towards the questions of social and institutional innovations. Hacking has not just further developed the technology itself but it has also created social innovations of informal co-operation and of licences sustaining creativity. Yet this is neither a natural consequence of technological development and progress nor an automatic result of the diverse cultures shaping the Internet. The technology has to be actively used for creative self-expression and developed in a direction supporting and extending such uses. In order to be socially sustainable, its self-referentiality in the sense of Castells needs to be complemented by reflexive action combining technical and social inventiveness. The technological possibilities need to be protected and sustained institutionally and socially as well: Through explicit subsidies by companies, governments and individuals donating webspace, time and expertise as well as through unofficial uses of organisational slack; through the discretion of software developers at work

and outside; through licences preventing commercial exploitation; and generally through actors' reflection upon the prerequisites and limitations of their own working conditions and a committment to extending the possibilities of participation and creative action.

Yet we have seen that on the level of developers' time use and involvement, FS/OS development is embedded with other activities: work and hobbyism, career investments and compensation for work frustrations figure, and frequently are articulated in complex ways – finding multiple uses for hard- and software, products and solutions, for professionalism and hedonism. Indeed, one of the not-so-secret secrets of FS/OS's success is that projects and tools are sufficiently modular, loosely coupled and transparent to allow for this kind of flexible involvement and heterogeneity and to technically support limited commitments.

FS/OS then represents more than an esoteric hobbyist community, but very likely less than a revolution. It embodies ways of thinking about the sustainability of chances and spaces for creative action and about the democratisation of technological and also cultural development (Holtgrewe, 2001; Stalder/Hirsh, 2002). However, its specific conditions suggest that FS/OS can hardly be considered as an all-embracing model of a mode of knowledge production. The prerequisites of its functioning may not be restricted to software development, but empirical evidence so far suggests that they are not by definition all-inclusive and do not extend over all parts of society. The picture of OS/FS then is a heterogeneous ensemble of technological, social and institutional innovations recursively enhanced by creative action in the sense of Castells, combined with limited and modular involvements, which nevertheless allow for more commitment and reflexivity than Kierkegaard, Dreyfus and Flores would have it.

With its very limitations and its context-specificity, our example suggests that theories of social and temporal differentiation can be usefully applied: "The" Internet appears to provide a wide range of both tools and media, and their uses, the articulations of actors, artefacts, institutions and practices both depend on and shape their respective contexts and arenas of action in diverse and unequal ways.

References

Adler, P. (2001) 'Towards Collaborative Interdependence: A Century of Change in the Organization of Work. Version of December 28th, 2001', unpublished manuscript.

Bauman, Z. (1997) *Postmodernity and Its Discontents*. Cambridge: Polity.

Beirne, M., Ramsay, H. and Panteli, A. (1998) 'Developments in Computing Work: Control and Contradiction in the Software Labour Process', in Thompson, P. and Warhurst, C. (eds) *Workplaces of the Future*, pp. 142-62. Houndsmills: Macmillan.

Benjamin, W. (1934) 'Der Autor als Produzent', *Gesammelte Schriften Bd II*, pp. 683-701. Frankfurt/Main: suhrkamp. = Benjamin, W. (1934/1999) 'The Author as Producer', in Jennings, M. W. (ed) *Walter Benjamin. Selected Writings Vol. 2*, pp. 220-32. Cambridge, Mass.: Harvard UP.

Benkler, Y. (2002) 'Coase's Penguin, or Linux and the Nature of the Firm', *Yale Law Journal* 112 http ://www.benkler.org/CoasesPenguin.pdf Version 04.1, downloaded 09/08/2002.

Bloch, E. (1962) *Erbschaft dieser Zeit. Gesamtausgabe Bd. 4*. Frankfurt/Main: suhrkamp. = Bloch, E. (1991) *Heritage of Our Times.* Berkeley: University of California Press.

Blutner, D., Holtgrewe, U. and Wagner, G. (1999) 'Charismatische Momente und Trajekte. Das Projekt als Plattform charismatischer Führung', in Schreyögg, G. and Sydow, J. (eds) *Führung - neu gesehen. Managementforschung 9*, pp. 199 - 237. Berlin/New York: de Gruyter.

Brecht, B. (1932) 'Der Rundfunk als Kommunikationsapparat. Rede über die Funktion des Rundfunks', *Bertolt Brecht Werke, Schriften I (1914-1933). Große Kommentierte Berliner und Frankfurter Ausgabe Bd. 21*, pp. 552-57. Frankfurt/Main: suhrkamp. = Brecht, B. (1932/2001) 'The Radio as a Communications Apparatus', in Silberman, M. (ed) *Brecht on Film and Radio*, pp. 40-48. London: Methuen.

Brose, H.-G. (2002) 'Kulturen der Ungleichzeitigkeit - Cultures of Non-Simultaneity.' Paper contributed to the KWI Conference "Speeding up Cultural Change" June 26 - 29, 2002 Mülheim

Castells, M. (1996) *The Rise of the Network Society.* Oxford: Blackwell.

Castells, M. (2001) 'Epilogue: Informationalism and the Network Society', in Himanen, P. (ed) *The Hacker Ethic and the Spirit of the Information Age*, pp. 155 - 78. London: Secker & Warburg.

Csikszentmihalyi, M. (1991) *Flow: The Psychology of Optimal Experience*. New York: Perennial.

De Certeau, M. (1988) *The Practice of Everyday Life.* Berkeley: University of California Press.

Debord, G. (1995) *Society of the Spectacle*. London: Zone Books.

Dreyfus, H. L. and Flores, F. (1997) 'Kierkegaard on the Information Highway: The Future of Information Technology as Seen in 1850 by Soren Kierkegaard' http://hci.stanford.edu/cs378/reading/dteyfus-kierkegaard.doc, downloaded 25/5/2002.

Edwards, K. (2001) 'Epistemic Communities, Situated Learning, and Open Source Software

Development.’Paper Presented to the "Epistemic Cultures and the Practice of Interdisciplinarity" Workshop at Nntu, Trondheim, June 11-12, 2001 http://www.its.dtu.dk/ansat/ke/ec_sl_oss.pdf, downloaded 25/5/2002.

Emirbayer, M. and Mische, A. (1998) 'What Is Agency?' *American Journal of Sociology* 103 (4): 962 - 1023

Esposito, E. (1993) 'Der Computer als Medium und Maschine', *Zeitschrift für Soziologie* 22 (5): 338-54

Esposito, E. (2002) *Soziales Vergessen. Formen und Medien des Gedächtnisses der Gesellschaft*. Frankfurt/Main: suhrkamp.

Ghosh, R. A., Robles, G. and Glott, R. (2002) 'Free/Libre and Open Source Software: Survey and Study Part V'. Maastricht: International institute of Infonomics, http://floss1.infonomics.nl/finalreport/FLOSSfinal-5.pdf, Download 25.7.2002.

Hacker, S. (1989) *Pleasure, Power and Technology. Some Tales of Gender, Engineering, and the Co-Operative Workplace*. Boston u. a.: Unwin Hyman.

Himanen, P. (2001) *The Hacker Ethic and the Spirit of the Information Age*. New York: Random House.

Holtgrewe, U. (2001) 'Kreativität als Norm - Zum Erfolg verdammt? Open-Source-Software zwischen sozialer Bewegung und technischer Innovation', in Allmendinger, J. (ed), *Gute Gesellschaft? Verhandlungen des 30. Kongresses der Deutschen Gesellschaft für Soziologie in Köln 2000*, pp. 399 - 424. Opladen: Leske.

Holtgrewe, U. and Werle, R. (2001) 'De-Commodifying Software? Open Source Software between Business Strategy and Social Movement', *Science Studies* 14 (1) 43 - 65

Horkheimer, M. and Adorno, T. W. (1971) *Dialektik der Aufklärung*. Frankfurt/Main: Fischer = Adorno, T. W. and Horkheimer, M. (1979) *Dialectic of Enlightenment*. London: Verso.

Iannacci, F. (2003) 'The Linux Managing Model' http://www.opensource.mit.edu.

Kalkowski, P. and Mickler, O. (2002) 'Zwischen Emergenz und Formalisierung - Zur Projektifizierung von Organisation und Arbeit in der Informationswirtschaft', *SOFI-Mitteilungen* (30): 119-34.

Knuth, D. E. (1972) *The Art of Computer Programming*. Reading, Mass.: Addison-Wesley.

Krafft, A. and Ortmann, G. (1988) 'Computer und Psyche. Angstlust am Computer'. Frankfurt/Main: Nexus

Lakhani, K. R., et al. (2002) 'The Boston Consulting Group Hacker Survey Release 0.73'

http://www.osdn.com/bcg/BCGHACKERSURVEY-0.73.PDF, download 5.8.02.

Lave, J. and Wenger, E. (1991) *Situated Learning - Legitimate Peripheral Participation*. Cambridge: Cambridge UP.

Lazzarato, M. (1998) 'Immaterielle Arbeit. Gesellschaftliche Tätigkeit unter den Bedingungen des Postfordismus', in Negri, T., Lazzarato, M. and Virno, P. (eds) *Umherschweifende Produzenten. Immaterielle Arbeit und Subversion*, pp. 39 - 52. Berlin: id-Verlag.

Lazzarato, M. (2002) *Videophilosophie. Zeitwahrnehmung im Postfordismus*. Berlin: b_books.

Moody, G. (2002) *Rebel Code. Linux and the Open Source Revolution, London*: Penguin.

Moon, J. Y. and Sproull, L. (2000) 'Essence of Distributed Work: The Case of the Linux Kernel', *First Monday* 5 (11) http:// www.firstmonday.org/issues/ issue5_11/moon/index.html, Download 12.11.2000.

Nonaka, I., Toyama, R. and Konno, N. (2001) 'SECI, Ba and Leadership: A Unified Model of Dynamic Knowledge Creation', in Nonaka, I. and Teece, D. J. (eds) *Managing Industrial Knowledge.* Creation, Transfer and Utilization, pp. 13-43. London: Sage.

Perlow, L. A. (1998) 'Boundary Control: The Social Ordering of Work and Family Time in a High-Tech Corporation', *Administrative Science Quarterly* 43: 328 – 57.

Rammert, W. (1997) 'New Rules of Sociological Method: Rethinking Technology Studies', *British Journal of Sociology* 48 (2): 171 –91.

Rasmussen, B. and Håpnes, T. (1991) 'Excluding Women from the Technologies of the Future? A Case-Study of the Culture of Computer Science. Ifim-Paper 10/91'. Trondheim: Institute for Social Research in Industry.

Raymond, E. (1998) 'The Cathedral and the Bazaar', *First Monday* 3 http://www.firstmonday.org/issues/issue3_3raymond/index.html, Download 25.7.2000.

Raymond, E. (1999) 'Shut up and Show Them the Code. Version 9.7.1999,' http://www.tuxedo.org/~esr/writings/shut-up-and-show-them.html, Download 25.7.2000.

Robles, G., et al. (2001) 'Who Is Doing It? A Research on Libre Software Developers, Version 27.8.2001,' http ://widi.berlios.de , figures http://widi.berlios.de/stats.php3.

Sennett, R. (1998) The Corrosion of Character: The Transformation of Work in Modern Capitalism. New York: Norton.

Stalder, F. and Hirsh, J. (2002) 'Open Source Intelligence', *First Monday* 7 (6): http://www.firstmonday.org/issues/issue7_6/stalder/index.html.

Strangleman, T. (2002) 'Speeding up Cultural Change and the Nostalgia for Permanence at Work.' Paper contributed to the KWI Conference "Speeding up Cultural Change" June 26 - 29, 2002 Mülheim.

Strübing, J. (1993) *Subjektive Leistungen im Arbeitsprozeß. Eine empirische Untersuchung von Arbeitsstilen in der Programmierarbeit*. Wiesbaden: DUV.

Tuomi, I. (2002) *Networks of Innovation. Change and Meaning in the Age of the Internet.* Oxford: Oxford UP.

Turkle, S. (1995) *Life on the Screen: Identity in the Age of the Internet*. New York: Simon & Schuster.

Turner, S. (1995) 'Charisma und Gehorsam.' *Berliner Journal für Soziologie* 5: 67 – 87.

Weick, K. (1995) *Sensemaking in Organizations.* Thousand Oaks, Calif. et al.: Sage.

Working Group on Libre Software (2000) 'Open Source: Information Society Opportunities for Europe? Version 1.2' http://eu.conecta.it/paper.pdf.

URSULA HOLTGREWE teaches sociology of work and organisation at Duisburg/Essen University. Institut für Soziologie, Universität Duisburg/Essen, D-47057 Duisburg, ++49-203-379-2739, fax ++49-203-379-4350, e-mail holtgrewe@uni-duisburg.de

---

[i] The complexity and erudition of Esposito's line of argument cannot be reconstructed here.

[ii] Sequential does not necessarily mean linear, though: Sensemaking may happen in fits and starts, by association or retrospection.

[iii] Technically, both terms mean roughly the same, but they mark an 'ideological' controversy and a difference in licencing policies: Free Software addresses creative freedom in an emphatic and normative sense, relating to "free speech, not free beer" (http://www.gnu.org/philosophy/free-sw.html). The term Open Source was coined in order to avoid that very connotation (Raymond, 1999)and focus on the technical quality of the code and development model. In the following, I shall use the abbreviation FS/OS.

[iv] This is also the reason why large, formerly commercial software packages which have been made open source such as the Mozilla web browser (ex-Netscape) or the OpenOffice office application (ex-StarOffice) take rather a long time to be taken up and developed further in the open source mode.

[v] Lakhani et al. surveyed a sample of 684 developers involved with a random selection of projects on Sourceforge who had been e-mailed and asked to participate personally. Both Robles et al. (2001, n=5478) and Ghosh et al. (2002, n=2784) put their questionnaire on the Internet and posted an invitation to participate to a selection of mailing lists and news forums.

[vi] This has been shown for the effects a predominant hacker/computer virtuoso culture has on the motivation of women students of computer science (Rasmussen/Håpnes, 1991).